Electronic Version
Stylesheet Version v1.1.1

Claims

[c1] What is claimed is:

1. A method for controlling a hardware circuit with a processor, the processor used for executing a code to control the hardware circuit, the code comprising:
a plurality of lower-level subroutines, wherein after the processor executes various lower-level subroutines, the hardware circuit will be controlled to execute various corresponding operations, and each lower-level subroutine will record results, which come from the hardware circuit executing the corresponding operations, in an error code; wherein each result corresponds to a recovery operation;
a plurality of higher-level subroutines, each higher-level subroutines used for calling at least a lower-level subroutine to control the hardware circuit to execute operations corresponding to the lower-level subroutine according to the called lower-level subroutine when the processor executesthe higher-level subroutine;
a plurality of recovery subroutines, each recovery subroutine corresponding to a recovery operations for controlling the hardware circuit to execute various corresponding recovery operations, after the processor exe-

cutes various recovery subroutines; and
an error-handling subroutine for calling the recovery subroutines according to the error code;
the method comprising:
after the processor executes the higher-level subroutines, executing the error-handling subroutine to allow the processor to control the hardware circuit to executethe corresponding recovery operations according to the results corresponding to the lower-level subroutines.

[c2] 2. The method of claim 1, wherein when the processor executes the error-handling subroutine after the higher-level subroutines are executed, the processor will not executesthe recovery operations corresponding to the lower-level subroutine until the higher-level subroutines are finished.

[c3] 3. The method of claim 1, wherein the higher-level subroutines won't call each other so that a next higher-level subroutine will not be executed until the processor finishes executing a previous higher-level subroutine.

[c4] 4. The method of claim 1, wherein the hardware circuit is a servo module of an optical storage drive, the servo module comprising:
a motor for driving an optical disk to rotate; and
a pick-up head for generating a laser incident on the op-

tical disk.

[c5] 5.The method of claim 1, wherein the hardware circuit is an interface module of an optical storage drive.

[c6] 6. The method of claim 1, wherein the error code is a global variable of the code; the operation results corresponding to the lower-level subroutines will be recorded in the same error code.

[c7] 7. The method of claim 1, wherein the code further comprises a plurality of next-level subroutines; when the processor executes various next-level subroutines, the hardware circuit is controlled to execute corresponding operations; each next-level subroutines will record operation results corresponding to the hardware circuitin a second error code;each lower-level subroutine is used for calling at least a next-level subroutine so that the processor sequentially executes the next-level subroutines of the lower-level subroutines to control the hardware circuit to execute corresponding operations when executing the lower-level subroutines.

[c8] 8. The method of claim 7, wherein the next-level subroutines of each lower-level subroutine record corresponding operation results in the same second error code.

[c9] 9. The method of claim 7, wherein the second error code is a column of the error code.

[c10] 10. The method of claim 7, wherein the next-level subroutines record corresponding operation results in the same second error code.

[c11] 11. The method of claim 7, wherein the second error code is a column of the error code.

[c12] 12. The method of claim 1, wherein the lower-level subroutines won't call each other so that a next lower-level subroutine will not be executed until the processor finishes executing a previous lower-level subroutine.

[c13] 13. The method of claim 1, wherein the lower-level subroutines won't call the higher-level subroutines.

[c14] 14. An electronic device, comprising:
a hardware circuit for achieving operations of the electronic device;
a processor for executing a code to control the hardware circuit;
a storage device for storing the code; wherein the code comprising:
a plurality of lower-level subroutines, wherein after the processor executes various lower-level subroutines, the

hardware circuit will be controlled to execute various corresponding operations, and each lower-level subroutine will record results, which come from the hardware circuit executing the corresponding operations, in an error code; wherein each result corresponds to a recovery operation;

a plurality of higher-level subroutines, each higher-level subroutines used for calling at least a lower-level subroutine to control the hardware circuit to execute operations corresponding to the lower-level subroutine according to the called lower-level subroutine when the processor executes the higher-level subroutine;

a plurality of recovery subroutines, each recovery subroutine corresponding to a recovery operations for controlling the hardware circuit to execute various corresponding recovery operations, after the processor executes various recovery subroutines; and

an error-handling subroutine for calling the recovery subroutines according to the error code;

wherein after executing the higher-level subroutines, the processor executes the error-handling subroutine to allow the processor to control the hardware circuit to execute the corresponding recovery operations according to the results corresponding to the lower-level subroutines.

[c15] 15. The electronic device of claim 14, wherein when the

processor executes the error-handling subroutine after the higher-level subroutines are executed, the processor will not executes the recovery operations corresponding to the lower-level subroutine until the higher-level subroutines are finished.

[c16] 16. The electronic device of claim 14, wherein the higher-level subroutines won't call each other so that a next higher-level subroutine will not be executed until the processor finishes executing a previous higher-level subroutine.

[c17] 17. The electronic device of claim 14 being an optical storage drive, the hardware circuit comprising a servo module, which comprising:
a motor for driving an optical disk to rotate; and
a pick-up head for generating a laser incident on the optical disk.

[c18] 18. The electronic device of claim 14 being an optical storage drive, the hardware circuit being an interface module of the optical storage drive.

[c19] 19. The electronic device of claim 14, wherein the error code is a global variable of the code; the operation results corresponding to the lower-level subroutines will be recorded in the same error code.

[c20] 20. The electronic device of claim 14, wherein the code further comprises a plurality of next-level subroutines; when the processor executes various next-level subroutines, the hardware circuit is controlled to execute corresponding operations; each next-level subroutines will record operation results corresponding to the hardware circuitin a second error code;each lower-level subroutine is used for calling at least a next-level subroutine so that the processor sequentially executes the next-level subroutines of the lower-level subroutines to control the hardware circuit to execute corresponding operations when executing the lower-level subroutines.

[c21] 21. The electronic device of claim 20, wherein the next-level subroutines of each lower-level subroutine record corresponding operation results in the same second error code.

[c22] 22. The electronic device of claim 20, wherein the second error code is a column of the error code.

[c23] 23. The electronic device of claim 20, wherein the next-level subroutines record corresponding operation results in the same second error code.

[c24] 24. The electronic device of claim 20, wherein the second error code is a column of the error code.

[c25] 25. The electronic device of claim 14, wherein the lower-level subroutines won't call each other so that a next lower-level subroutine will not be executed until the processor finishes executing a previous lower-level subroutine.

[c26] 26. The electronic device of claim 14, wherein the lower-level subroutines won't call the higher-level subroutines.